LoRaWAN

LoRaWAN
Version 1.0

Hello, and welcome to this presentation of the Low Power Wide Area networking protocol.

LoRaWAN overview

- LoRa® technology:
  - uses license-free spectrum below 1GHz like 868MHz for Europe and 915MHz for North America.
  - Long-range transmissions (>10kms)
  - Low power consumption
- LoRaWAN® protocol is:
  - Bi-directional
  - Simple or Half duplex (with optional Full duplex)
  - Modulation LoRa (Chirp spread spectrum) and FSK

LoRaWAN protocol has an official regional specification for radio frequency usage (frequency range, default parameters, limitations, …)

life.augmented

The LoRa® technology is an Internet of Things solution for connecting many devices to some wide-area networks.
It's defined as an open global standard to ensure compliance and ability to build your infrastructure offering.

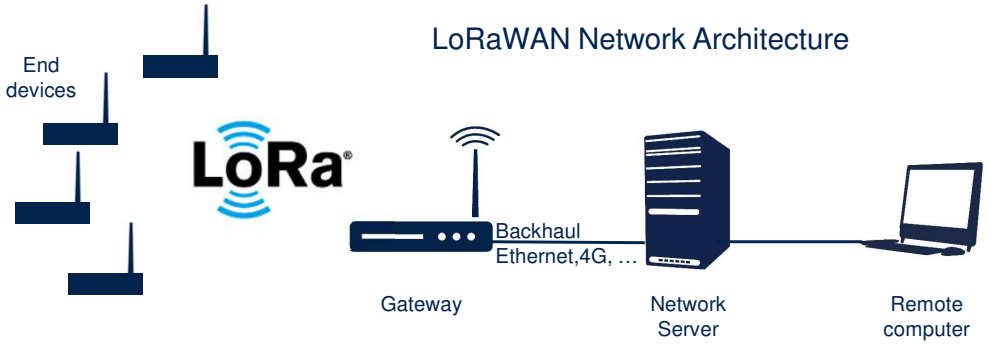
Network (1/2)

- LoRa technology is able to provide a wide area network capability, it is often referred to as **LoRaWAN**.
- A LoRaWAN network consists of several elements:
  - ***End devices (aka end nodes):*** The end devices are the elements of the LoRaWAN network where the sensing or control is undertaken. They are normally remotely located.
  - ***Gateway :*** The gateway receives the communications from the LoRaWAN end devices and then transfers them onto the backhaul system. This part of the LoRaWAN network can be Ethernet, cellular or any other telecommunications link wired or wireless. The gateways are connected to the network server using standard IP connections. On this way the data uses a standard protocol, but can be connected to any telecommunications network, whether public or private. In view of the similarity of a LoRaWAN network to that of a cellular one, LoRaWAN gateways may often be co-located with a cellular base station. In this way they are able to use spare capacity on the backhaul network.
  - ***Server:*** The LoRaWAN network server manages the network. The network server acts to eliminate duplicate packets, schedules acknowledgement, and adapts data rates. In view of the way in which it can be deployed and connected, makes it very easy to deploy a LoRaWAN network.
  - ***Remote computer:*** a remote computer can then control the actions of the end devices or collect data from them - the LoRaWAN network being almost transparent.

life.augmented

Each end device can send and receive information in order to provide sensor data or to control devices via private or public networks in a secure and certified way.

Network (2/2)



- In terms of the actual architecture for the LoRaWAN network, the end devices are typically in a star-of-stars topology with gateways forming a transparent bridge. These relay messages between end devices and a central network server in the backend.
- Communication to end devices is generally bi-directional, but it is also possible to support multicast operation, and this is useful for features such as software upgrades and the like or other mass distribution messages.

life.augmented

4

LoRaWAN networks typically are laid out in a star-of-stars topology in which gateways relay messages between end devices and a central network server at the backend. Gateways are connected to the network server via standard IP connections while end devices use single-hop LoRa™ or FSK communication to one or many gateways. All communication is generally bi-directional, although uplink communication from an end device to the network server is expected to be the predominant traffic.
Communication between end devices and gateways is spread out on different frequency channels and data rates.

Region overview

- 10 Regions are defined:
  - EU868 (863-870MHz) *
  - US915 (902-928MHz) *
  - CN779 (779-787MHz)
  - EU433 (433-434MHz)
  - AU915 (915-928MHz)
  - CN470 (470-510MHz)
  - AS923 (915-928MHz) *
  - KR920 (920-923MHz) *
  - IN865 (865-867MHz) *
  - RU864 (864-870MHz)
- (*) The LoRaWAN Certification^{CM} Program provides a suite of regional tests in 5 regions to validate that application-specific end devices operate on any LoRaWAN network.
- For each region:
  - The network channels can be freely attributed by the network operator in compliance with the allowed sub-bands defined.
  - The Physical bitrates (Data Rate), Tx Power ranges, and default settings are defined.
  - Some restrictions can be defined (Duty Cycle, Dwell Time, Listen Before Talk)

life.augmented

5

The LoRaWAN protocol defines 10 main regions as references. Some default parameters are provided for different regulatory regions worldwide as default frequency channels, Tx Power or Data rates range.
For each country, one or more channel plans have been defined in order to ensure the consistency of the rules in force in each territory.

LoRaWAN protocol

LoRaWAN classes

- ***Class A - bi-directional end-devices:*** LoRaWAN Class A provides bidirectional communications. To achieve this, each end-device transmission is followed by two short downlink receive windows. The transmission slot scheduled by the end device is based upon the needs of the end device with a small variation determined using a random time basis.
  LoRa Class A operation provides the lowest power option for end devices that only require downlink communication from the server shortly after the end device has sent an uplink transmission. Downlink communications from the server at any other time wait until the next scheduled uplink.
- ***Class B - bi-directional end-devices with scheduled receive slots:*** LoRaWAN Class B provides the Class A functionality and, in addition to this, they open extra receive windows at scheduled times. To achieve the required synchronization from the network, the end device receives a time synchronized Beacon from the gateway. This allows the server to know when the end device is listening.
- ***Class C - bi-directional end-devices with maximal receive slots:*** LoRaWAN Class C provides nearly continuously open receive windows. They only closed when the end device is transmitting. This type of end device is suitable where large amounts of data are needed to be received rather than transmitted.

life.augmented

6

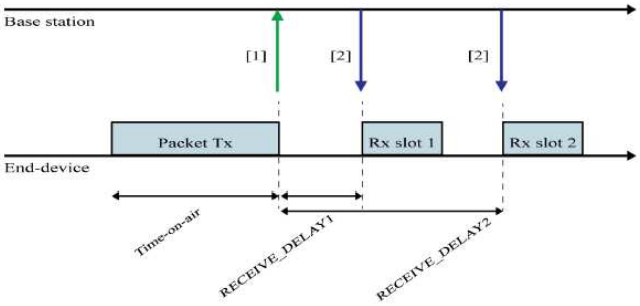
A LoRaWAN network distinguishes between a basic LoRaWAN (named Class A) and optional features (Class B, Class C):

- Class A: End devices of Class A allow for bi-directional communications whereby each end-device's uplink transmission is followed by two short downlink receive windows.
- Class B: End devices of Class B allow for more receive slots by extra scheduled times receive windows.
- Class C: End devices of Class C have almost continuously open receive windows.

LoRaWAN protocol

Class A management

- Class A: when the end device wants to send message (sensors data, MAC command, …), an uplink transmission is sent to network server relayed by one or many gateways.
  Network can respond (Acknowledge message, actuator data, MAC command) during downlink Rx1 or Rx2 reception windows in using a single gateway.
  An end device does not open Rx2 if a downlink frame is received in Rx1.



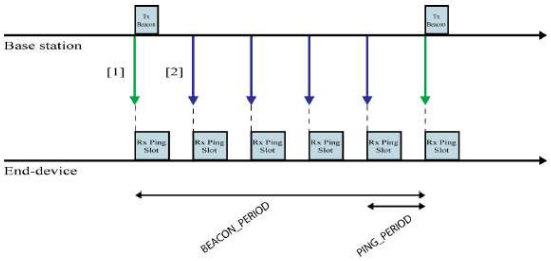
life.augmented

Regarding Class A, when the end device wants to send message (sensors data, MAC command, …), an uplink transmission is sent to network server relayed by one or many gateways.
Network can respond (Acknowledge message, actuator data, MAC command) during downlink Rx1 or Rx2 reception windows in using a single gateway.
An end device does not open Rx2 if a downlink frame is received in Rx1.

LoRaWAN protocol

Class B management

- Class B: adds some synchronized reception windows on the end device.
  LoRaWAN end device joins always in Class A and might switch to Class B upon application layer request.



life.augmented

Regarding Class B, the end device adds some synchronized reception windows.
LoRaWAN end device joins always in Class A and might switch to Class B upon application layer request.

LoRaWAN protocol
Class B management

- Principle of synchronization for end device Class B
  - End device always starts and Joins network as Class A end device (Class B capable/disabled)
  - Class B enabled request always comes from Application layer of the end device
  - Switch to Class B enabled
    - End device search for Beacon in order to lock on it
      - To accelerate beacon discovery a “DeviceTimeReq” MAC cmd can be used
    - When Locked, the end device opens receive windows (called “ping slot”)
      - To operate in Class B, Ping-Slot information shall be made available to the network (Periodicity, Data Rate and frequency)
    - Once Class B mode, the MAC layer shall set to 1 the “Class B” bit of the FCtrl field of each uplink frame transmitted.

life.augmented

End devices should implement Class B operation, there is a requirement to open the reception windows at fixed time intervals.
The decision to switch from Class A to Class B comes from the application layer of the end device.
The beacon frame is used to synchronize end devices and gateways on the same timestamp. To be able to switch to Class B, the device must release at least one beacon.

LoRaWAN protocol

Class B management

- During Class B mode operation:
  - Server may change the end device's ping slot downlink frequency or data rate thanks to PingSlotChannelReq MAC command.
  - End device may change the periodicity of its ping slots thanks to PingSlotInfoReq MAC command.
  - If the end device requires to send a Class A uplink message, and during the Tx/Rx1/Rx2 time frame, the Class A configuration will take priority. All messages sent by the server on the ping slot window during this time frame will be lost.
- The Class B messages can be unicast (sent to a single end device) or multicast (sent to multiple end devices).
  If the 2 ways are enabled at the same time but with some different parameters, the application layer must decide which one must be used to configure the ping slot windows.

life.augmented

The Class B configuration is defined by the Regional Parameters with some default settings as ping slot frequency and data rate. All these values can be updated through some MAC command instructions from the network server.
In addition, the Class B ping slots can be used as Unicast or Multicast windows with an optional Multicast configuration defined by the network server.

LoRaWAN protocol

Class B management

- Beacon window is usable time interval for Class B slots

BEACON_PERIOD	128 s
BEACON_RESERVED	2.120 s
BEACON_GUARD	3.000 s
BEACON_WINDOW	122.88 s

- Ping slot timing calculation for every beacon window period

K	0	1	2	3	4	5	6	7
Ping Nb (= 2^{K})	1	2	4	8	16	32	64	128
Ping Period (= $2^{(12-K)}$)	4096	2048	1024	512	256	128	64	32
Interrupt period (s)	122.88	61.44	30.72	15.36	7.68	3.82	1.92	0.96

- At each beacon period a slot index (pingOffset) is randomized and changed (avoid collisions)

First slot	BEACON_RESERVED + PingOffset x slotLen (30ms)
Slot 2	BEACON_RESERVED + (PingOffset + PingPeriod)x slotLen (30ms)
Slot 3	BEACON_RESERVED + (PingOffset + (2 x PingPeriod)) x slotLen (30ms)
……	…….

life.augmented

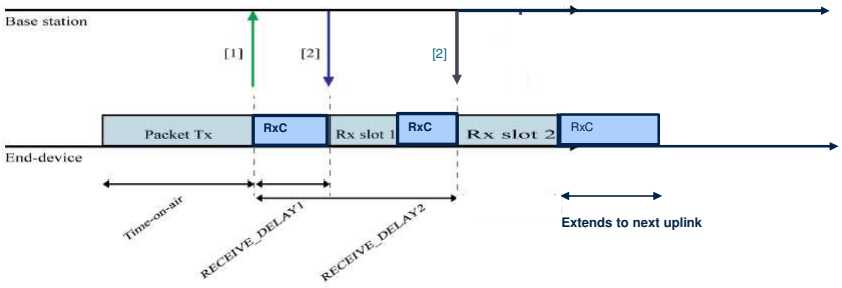
11

The beacon generated by the gateways is sent every 128s. All end devices with the Class B mode enabled must open a listening window to receive this frame.
As defined by the Class B configuration between the end device and the network server, the ping slot window is configured to be opened between once per second or once every 128 seconds.
In addition, the ping slot calculation uses a pseudo-random offset to align receive windows and avoid collisions with multiple end devices.

LoRaWAN protocol

Class C management

- Class C: adds a synchronized continuously reception window on the end device.
  LoRaWAN end device joins always in class A and might switch to class C upon application layer request.



life.augmented

12

Class C end devices add a synchronized continuously reception window.
LoRaWAN end device joins always in class A and might switch to class C upon application layer request.

LoRaWAN protocol
Class C management

- End device Class C capable/enable
  - End device Class B and Class C capable shall not enable Class B and Class C in concurrent mode.
  - End device Class C enabled listens using a channel/DR parameters combination referred to as RxC as often as possible.
  - End device SHALL listen on RxC when it is not either sending or receiving on Rx1 or Rx2, according to Class A definition.
- Rx priority during Class C mode Operation
  - If the end device requires to send a Class A uplink message, and during the Tx/Rx1/Rx2 time frame, the Class A configuration will take priority. All messages sent by the server on the RxC window during this time frame will be lost.
  - As an end device does not open Rx2 if a downlink frame is received in Rx1, the end device opens immediately a continuous RXC.
- RxC parameters are identical by default to the Rx2 parameters (same channel and same data rate).
- The Class C messages can be unicast (sent to a single end device) or multicast (sent to multiple end devices). If the 2 ways are enabled at the same time but with different parameters, the application layer must decide which one must be used to configure the RxC windows.

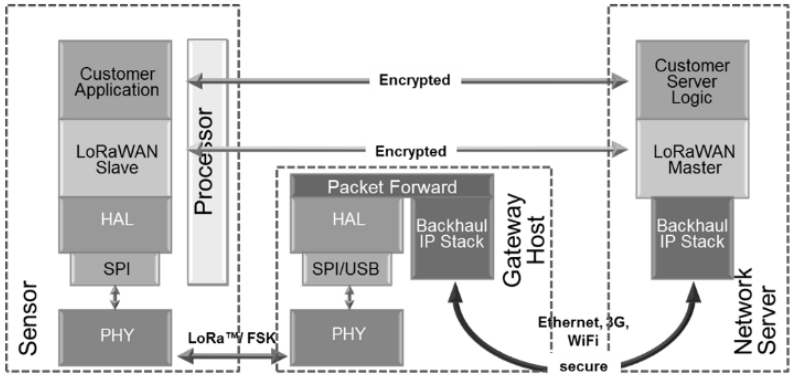
life.augmented

If a device can operate in Class B and C, only one of these Classes can be activated at a time, in addition to Class A.
The Class C configuration is set by default with the same parameters as Rx2.
As for Class B, the RxC windows can be used to receive Unicast or Multicast messages, thanks to an optional Multicast configuration defined by the network server.

LoRaWAN protocol

Protocol stack overview

- The LoRaWAN protocol stack is defined with 3 layers:
  - Application layer
  - MAC layer (network)
  - Physical layer
- LoRaWAN implements several keys, which may be unique to each end device, in order to ensure the security of exchanges at the network and application level between the end device and the network.



life.augmented

The LoRaWAN protocol stack is defined with 3 layers:
- Application layer
- MAC layer (network)
- Physical layer

LoRaWAN implements several keys, which may be unique to each end device, in order to ensure the security of exchanges at the network and application level between the end device and the network.

LoRaWAN protocol

LoRa network security

- The confidentiality of LoRaWAN messages is protected by AES-128 encrypting the FRMPayloadfield. LoRaWAN differentiates between MAC messages destined for the network server and application messages destined for the application server through the FPort field (which is not encrypted), thus two different encryption keys are used for the FRMPayload field depending on the intended destination.
- End devices in LoRaWAN thus needs two 128-bit encryption keys, one Network Session Key (NwkSKey) for encrypting MAC messages and one Application Session Key (AppSKey) for encryption application messages.
- The **AppSKey** is an encryption key shared between an end device and the application server, while the **NwkSKey** is an encryption key shared between the end device and the network server.

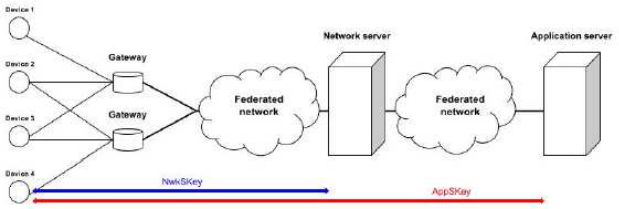


Figure 2.5: The LoRaWAN architecture illustrating how the AppSKey and the NwkSKey are shared and used.

life.augmented

15

This security is ensured thanks to two AES 128-bit session keys in order to encrypt network layer messages (MAC) and application layer messages.
These keys are either provided to the end device during programming or generated through a join step with intermediate keys.

LoRaWAN protocol

LoRa network security

- In order to provide message integrity, a Message Integrity Code (MIC) is computed for each packet using AES-CMAC and the NwkSKey. The MIC is computed over the entire PHYPayload (after the **FRMPayload** has been encrypted) in order to detect packet tampering.

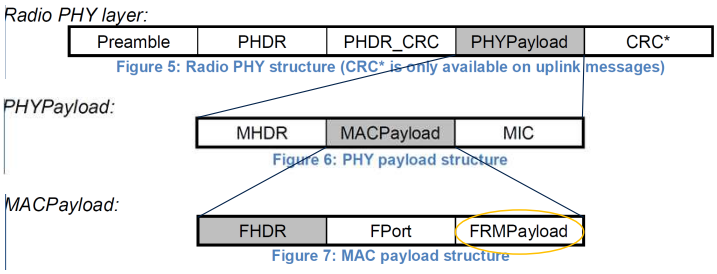
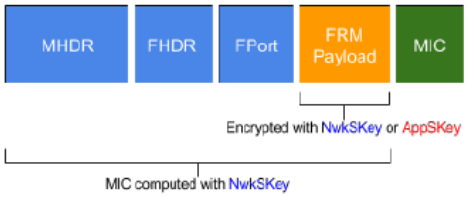


Figure 5: Radio PHY structure (CRC* is only available on uplink messages)

Figure 6: PHY payload structure

Figure 7: MAC payload structure



life.augmented

16

In order to provide message integrity, a Message Integrity Code (MIC) is computed for each packet using AES-CMAC and the NwkSKey. The MIC is computed over the entire PHYPayload (after the FRMPayload has been encrypted) to detect packet tampering.

LoRaWAN protocol

LoRa network security (commissioning)

- OTAA (Over-the-air activation)
  - Commissioning parameters in OTAA:
    - DevEUI of an end device has a similar function to that of a MAC-address in WiFi (ID to identify in a unique way the end device).
    - the AppEUI (JoinEUI) has a similar function to the SSID in WiFi (ID to identify in a unique way the network server).
    - The AppKey in OTAA has a similar function to that of a password in WiFi (secure application Key).
  - The OTAA uses an over the air message handshake that is carried out as follows
    - The end device transmits a **join-request message** to the network server (indicated by setting the MType field in the MHDR to 000). The join-request message is not encrypted but it is protected by the **MIC**, in this case calculated with the Application Key (AppKey), which is an AES-128 key. A join-request contains:
      - Globally unique end-device Identifier (DevEUI)
      - Application Identifier (AppEUI)
      - A nonce of 2 octets (DevNonce). The DevNonce is a random value that the network server keeps track of for each end device.
  - The end device receives a **join-accept message** (indicated by the MType field in the MHDR set to 001) from the network server if the end device is permitted to join the network.

life.augmented

17

To obtain these keys, there are 2 methods to activate the LoRaWAN end device: OTAA or ABP.
The OTAA mode requires an element to uniquely identify the end device: the DevEUI.
The end device must also know the network identity and the network key to connect to, via the AppEUI and the AppKey.
This step is defined as a Join Request. If the network receives a request from an end device and accepts it, the end device can start exchanging encrypted messages using the sessions keys created by the Join.

LoRaWAN protocol

LoRa network security (commissioning)

- ABP (Activation by personalization)
  - The alternative way of joining a LoRaWAN network and sharing keys is the Activation By Personalisation (ABP) procedure.
  - ABP relies on using pre-shared network and application keys, mainly stored at production time.
    - Device Address (DevAddr)
    - Network Session Key (NwkSKey)
    - Application Session Key (AppSKey)

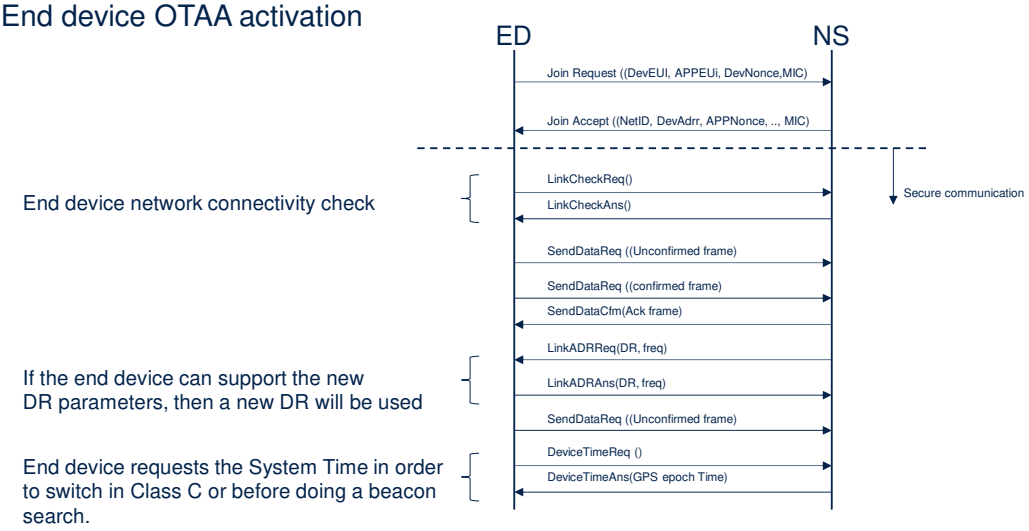
life.augmented

18

The ABP mode uses pre-defined session keys between the network and the end device. These keys must be programmed during the production phase of the end devices.
At this moment, the device can start immediately by exchanging encrypted messages via these private keys.
In summary, OTAA method is more complex to implement than ABP method but offers a better level of security. Indeed, in ABP the session keys are statics whereas in OTAA they are derived at each Join-Request session.

LoRaWAN protocol
Message Sequence Chart

- End device OTAA activation



life.augmented

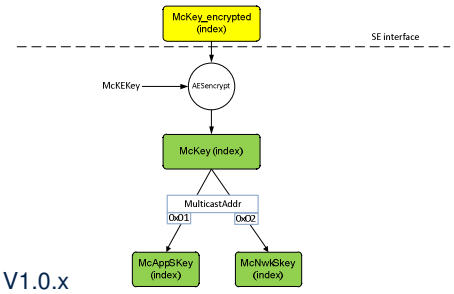
The sequence diagram below shows the end device startup in OTAA mode, with a series of MAC and application messages. The MAC messages are used to configure the end device according to network recommendations/constraints.

LoRaWAN protocol

Multicast key derivation

- How to derive the McAppSKey and McNetSkey from the McKey ?
  - Explain the key derivation and do distinction between unicast and multicast key.
  - During a Multicast setup from application layer the end device will receive a McKey_encrypted (it is the encrypted McKey)
  - Flow to get back the McKey and session key computation
    - **McKey** = aes128_encrypt(McKEKey, McKey_encrypted)
    - McRootKey = aes128_encrypt(**GenAppKey**, 0x00 | pad_{16})
    - McKEKey = aes128_encrypt(McRootKey, 0x00 | pad_{16})

      Finally
    - McAppSKey = **aes128_encrypt(McKey**, 0x01 | McAddr | pad_{16})
    - McNetSKey = **aes128_encrypt(McKey**, 0x02 | McAddr | pad_{16})
    - **GenAppKey** is a new root key provisioned in the end device LoRaWAN V1.0.x



life.augmented

Multicast is a communication overlay available with the Class B or Class C usage.
In addition to the native session keys for the exchange of unicast encrypted messages between the end devices and the network, there are session keys for multicast messages.
These keys are generated during a setup step of the Multicast group between an application server and several end devices.
Here are the different steps to generate these keys.

Medium Access Control

MAC Message formats

- **MHDR** (MAC header) specifies the message type:
  - Join Request/Accept messages
  - Data up/down messages. They are used to transfer both MAC commands and application data.
- **FHDR** (Frame header) contains the DevAddr, adaptative rate control, frame counter, Acknowledgement and MAC commands if any.
- **FPort** is between 1 and 223 if application-specific or 0 if the FRMPayload contains only MAC commands.
- **FRMPayload** is the data frame (App or MAC).
- **FCtrl** is frame control and **FOpts** is the frame options used to transport MAC commands.
- The message integrity code (**MIC**) is calculated over all the fields in the message (AES).

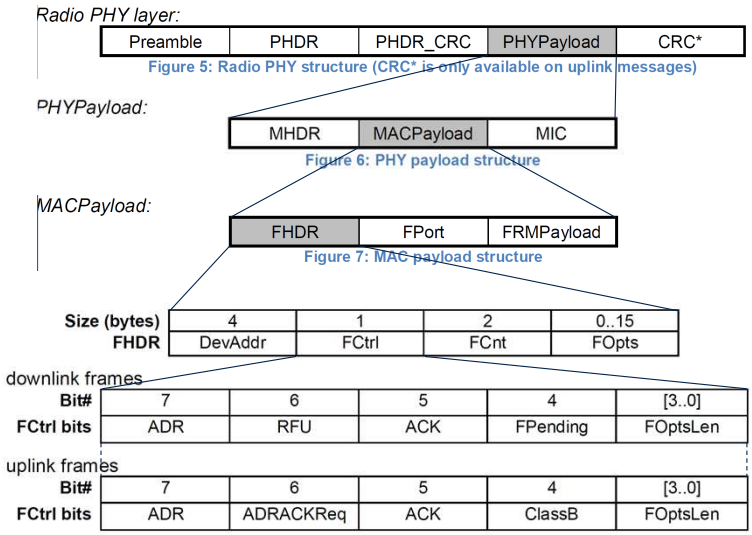


Figure 5: Radio PHY structure (CRC* is only available on uplink messages)

Figure 6: PHY payload structure

Figure 7: MAC payload structure

21

LoRaWAN uplink and downlink messages carry a Physical payload starting with a MAC header, followed by a MAC payload, and ending with a message integrity code (MIC).
The MAC payload contains a frame header followed by a port field and an optional frame payload.
This MAC payload is used to exchange MAC commands, controls parameters, as well as a data frame, in the case of an uplink from the application layer.

Medium Access Control

MAC commands

- For network administration, a set of MAC commands may be exchanged exclusively between the network server and the MAC layer on an end device. MAC layer commands are never visible to the application server or the application running on the end device.

CID	Command	Transmitted by: End-device	Transmitted by: Gateway	Short Description
0x02	***LinkCheckReq***	x		Used by an end-device to validate its connectivity to a network.
0x02	***LinkCheckAns***		x	Answer to LinkCheckReq command. Contains the received signal power estimation indicating to the end-device the quality of reception (link margin).
0x03	***LinkADRReq***		x	Requests the end-device to change data rate, transmit power, redundancy, or channel mask.
0x03	***LinkADRAns***	x		Acknowledges the LinkADRReq.
0x04	***DutyCycleReq***		x	Sets the maximum aggregated transmit duty-cycle of a device.
0x04	***DutyCycleAns***	x		Acknowledges a DutyCycleReq command.
0x05	***RXParamSetupReq***		x	Sets the reception slots parameters.
0x05	***RXParamSetupAns***	x		Acknowledges a RXParamSetupReq command.
0x06	***DevStatusReq***		x	Requests the status of the end-device.
0x06	***DevStatusAns***	x		Returns the status of the end-device, namely its battery level and its radio status.
0x07	***NewChannelReq***		x	Creates or modifies the definition of a radio channel.
0x07	***NewChannelAns***	x		Acknowledges a NewChannelReq command.
0x08	***RXTimingSetupReq***		x	Sets the timing of the of the reception slots.
0x08	***RXTimingSetupAns***	x		Acknowledges RXTimingSetupReq command.
0x09	***TXParamSetupReq***		x	Used by the Network Server to set the maximum allowed dwell time and Max EIRP of end-device, based on local regulations.
0x09	***TXParamSetupAns***	x		Acknowledges TXParamSetupReq command.
0x0A	***DlChannelReq***		x	Modifies the definition of a downlink RX1 radio channel by shifting the downlink frequency from the uplink frequencies (i.e. creating an asymmetric channel).
0x0A	***DlChannelAns***	x		Acknowledges DlChannelReq command.
0x0B to 0x0C	***RFU***			
0x0D	***DeviceTimeReq***	x		Used by an end-device to request the current GPS time
0x0D	***DeviceTimeAns***		x	Sent by the Network Server, answer to the DeviceTimeReq request
0X0E to 0x7F	RFU			
0x80 to 0xFF	Proprietary	x	x	Reserved for proprietary network command extensions

life.augmented

22

For network administration, a set of MAC commands may be exchanged exclusively between the network server and the MAC layer on an end device. MAC layer commands are never visible to the application server or the application running on the end device.

Medium Access Control
MAC commands

- The Class B specification adds following MAC commands.
  All commands described in the previous slide (Class A specification) SHALL be implemented in Class B capable end devices.

CID	Command	Transmitted by		Short Description
		End-device	Gateway	
0x10	***PingSlotInfoReq***	x		Used by the end-device to communicate the unicast ping slot periodicity to the Network Server
0x10	***PingSlotInfoAns***		x	Used by the network to acknowledge a PingInfoSlotReq command
0x11	***PingSlotChannelReq***		x	Used by the Network Server to set the unicast ping channel frequency and data rate of an end-device
0x11	***PingSlotChannelAns***	x		Used by the end-device to acknowledge a ***PingSlotChannelReq***command
0x12	***BeaconTimingReq***	x		Deprecated
0x12	***BeaconTimingAns***		x	deprecated
0x13	***BeaconFreqReq***		x	Command used by the Network Server to modify the frequency at which the end-device expects to receive beacon broadcast
0x13	***BeaconFreqAns***	x		Used by the end-device to acknowledge a BeaconFreqReq command

life.augmented

The Class B specification adds following MAC commands.
All commands described in the previous slide (Class A specification) SHALL be implemented in Class B capable end devices.

Medium Access Control
MAC commands

- A MAC command consists of a command identifier (CID) of 1 byte followed by a possibly empty command-specific sequence of bytes.
- Examples of MAC command for Class A
  - LinkADRReq/Ans (network -> end device): CID = 0x03
    - Allows a network to request an end device to perform a data rate adaptation on uplink frame transmission.
      A ChannelMask field proposes the channels usable for uplink transmission and the Redundancy field proposes the “NbTrans” for each uplink message (only applies on “unconfirmed” uplink frame).
    - The response message from the end device informs the network server if the configuration has been accepted.

Size (bytes)	1	2	1
LinkADRReq Payload	DataRate_TXPower	ChMask	Redundancy

Bits	[7:4]	[3:0]
DataRate_TXPower	DataRate	TXPower

Size (bytes)	1
LinkADRAns Payload	Status

Bits	[7:3]	2	1	0
Status bits	RFU	Power ACK	Data rate ACK	Channel mask ACK

  - DeviceTimeReq/Ans (end device -> network): CID = 0x0D
    - Allows an end device to request from the network the current network time in GPS epoch format.
      The end device can synchronize its internal clock to the network’s clock. Useful to speed-up the Class B beacon acquisition.

life.augmented

24

A MAC command request is defined by one byte for CID value, a unique command identifier, followed by an optional command-specific sequence of bytes. This request may be transmitted by the end device or by the network server, depending on the type of message to transmit.
By example, the LinkADR request is sent from the network server to the end device to update the data rate, TX power or the channels to use.
And the DeviceTime request is sent from the end device to the network server to synchronize the internal clock to the network time.

Medium Access Control
MAC commands

- Examples of MAC commands for Class B
  - PingslotInfoReq/Ans (end device -> network): CID = 0x10
    - Allows an end device to inform the network of its ping slot periodicity.
      This request may only be used to inform the server of a UNICAST ping slot. The Multicast ping slot is defined by the application server and will not use this command. The Multicast ping slot periodicity can be provided by the McGroupSetupReq, with an additional package specification.
  - PingSlotChannelReq/Ans (network -> end device): CID = 0x11
    - Allows a network to send to an end device a request to modify the frequency and/or the data rate of the downlink ping slot.
      This request can only be received in Class A mode (on Rx1 or Rx2 windows).

life.augmented

Several optional MAC commands are available when the end device is Class B capable.
These instructions are useful to manage the ping slot or beacon configurations with an update of the frequency, the data rate or periodicity.

Medium Access Control

Data rate adaptation

- Data rate adaptation on retransmission messages
  - When an end device sends a “confirmed” uplink frame toward the network, it expects to receive an acknowledgement from the network, in using the ACK bit of the FCtrl in the next subsequent Rx slot. If it does not receive this “ack” bit, then it tries to re-transmit the same data again.
  - The re-transmission can happen either on a new frequency or also can happen at a different data rate (preferable lower) than the previous one. A sending recommended strategy to adopt will be:
    - First “confirmed” uplink frame is sent with the data rate “DR” and the next retransmissions (in case of) will follow the rule table in below.
    - If after the 8 transmissions, the message has not been acknowledged then the MAC layer returns an execution error to the application layer.
    - For each retransmission, the frequency channel is randomly selected as standard transmissions.

Trans Nb	Data Rate
1 (first)	DR
2	DR
3	Max(DR-1,0)
4	Max(DR-1,0)
5	Max(DR-2,0)
6	Max(DR-2,0)
7	Max(DR-3,0)
8	Max(DR-3,0)

life.augmented

In the case of highly noisy signals resulting in loss of messages, the MAC layer uses retransmission procedures with data rate adaptation.

The data rate is a parameter to slow down or accelerate the transmission rate in order to best react to the environment and the distance between the end device and the gateway.

LoRaWAN regional phyLayer

Duty-cycle, LBT, dwell time

- The LoRaWAN specify some rules to be able to communicate between end devices and gateways:
  - The end device changes channel in a pseudo-random fashion for every transmission.
  - The end device respects the maximum transmit duty cycle relative to the sub-band used and local regulations.
  - The end device respects the maximum transmit duration relative to the sub-band and local regulations.
- With a duty-cycled local regulation, each time a frame is transmitted on a given sub-band, the time of emission and the on-air duration of the frame are recorded for this sub-band. The same sub-band cannot be used again during the next Toff seconds. During the unavailable time of a given sub-band, the device may still be able to transmit on another sub-band.If all sub-bands are unavailable due to duty-cycle limitation, the device can no longer send a message, until at least one element is available again.
- The Toff is calculated with the below equation:

$$Toff_{subband} = \frac{TimeOnAir}{DutyCycle_{subband}} - TimeOnAir$$

- By example, the EU868 ISM band defines a per sub-band duty-cycle limitation of 1% to comply with the ETSI regulations. With a 1% duty-cycle and Tx transmit time of 500ms, the used sub-band is unavailable during 49.5sec.
- Others transmission management are defined as the Listen Before Talk on KR920 ISM band, consisting in starting the transceiver by listening before using a frequency. The Dwell Time is also a regulation on the size of the transmitted data.

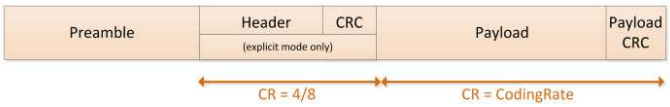
life.augmented

27

There is a set of rules applicable to all LoRaWAN end devices, but there are also rules that apply only to the locality according to the regulatory authorities.
By example, the EU868 ISM band defines a 1% duty-cycle for end devices transmissions, to prevent overloading of the frequencies used.
Others transmission management are defined as the Listen Before Talk on KR920 ISM band, consisting in starting the transceiver by listening before using a frequency. The Dwell Time is also a regulation on the size of the transmitted data.

LoRa Physical layer

- LoRa modem uses the physical (Phy) packet structure defined below:



  - **Preamble** is used to synchronize receiver with the incoming signal. The preamble format is specific for each ISM Band. In the LoRaWAN regional parameters specifications, the default public LoRa modulation uses an 8 symbols length with a sync word of 0x34.
  - **Header** provides information on the payload:
    - The payload length in bytes
    - The forward error correction code rate
    - The presence of an optional 16-bits CRC for the payload
  - **Payload** contains the MAC Header, MAC Payload and a MAC integrity code compatible with 802.15.4 recommendations. If the preamble or the header doesn't contain the expected information, the physical layer doesn't transmit the payload to the MAC layer.
  - **CRC** only available on uplink messages.

life.augmented

28

The LoRa packet starts with a preamble sequence which is used to synchronize the receiver with the incoming signal. By default, the packet is configured with an 8-symbol long sequence.
The preamble is followed by a header which contains information about the following payload. The packet payload is a variable-length field that contains the actual data coded at the error rate either as specified in the header in explicit mode or as selected by the user in implicit mode. An optional CRC may be appended.

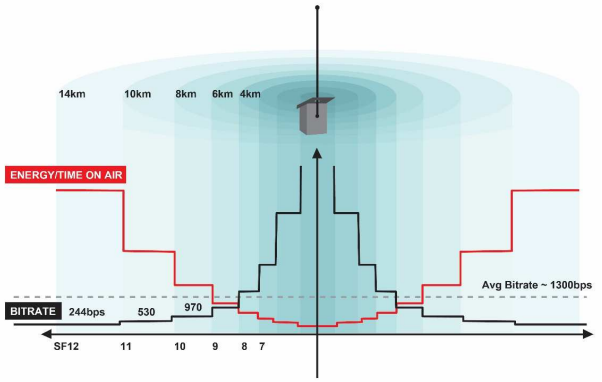
LoRa Physical layer

- LoRa modem uses spread spectrum modulation and forward error correction techniques to increase the range and robustness or radio communication:
  - The bandwidth permits the use of a higher effective data rate, thus reducing transmission time. (from 125KHz to 500KHz)
  - The spreading factor is the number of symbols sent per bit of information. (from SF5 to SF12)
  - The coding rate is the cycle error coding rate to perform forward error detection and correction (FEC). (from 4/5 to 4/8)
- The Time-on-Air calculation defines the time usage of the transceiver:

$$ToA = \frac{2^{SF}}{BW} * N_{symbols}$$

- Time-on-Air tool calculator:

  https://www.loaratools.nl/#/airtime



life.augmented

29

An important facet of the LoRa modem is its increased immunity to interference. This immunity to interference permits the simple coexistence of LoRa modulated systems either in bands of heavy spectral usage or in hybrid communication networks that use LoRa to extend range when legacy modulation schemes fail.
It is possible to optimize the LoRa modulation for a given application, access is given to the designer to some critical design parameters, each one permitting a trade-off between the link budget, immunity to interference, spectral occupancy and nominal data rate.